Automated Parameter Selection Tool for Solution to Ill-Posed Problems
Proposal

Brianna Cash (brcash@math.umd.edu)

Advisor: Dianne O'Leary (oleary@cs.umd.edu)

September 21, 2011

Abstract

In many ill-posed problems it can be assumed that the error in the data is dominated by noise which is independent identically normally distributed. Given this assumption the residual should also be normally distributed with similar mean and variance. This idea has been used to develop three statistical diagnostic tests to constrain the region of plausible solutions. This project aims to develop software that automates the generation of a range of plausible regularization parameters based on diagnostic tests.

1 Background

In medical images such as MRI or CT scans, the images may be distorted and/or noisy due to the physics of the measurement and the structure of the material (human) being imaged. These images are expensive to produce and often are critical in making medical decisions. Image blurring is an example of an ill-posed inverse problem. To find suitable approximate solutions to ill-posed inverse problems we use our knowledge about the particular problem to come up with constraints [4]. These constraints are used to determine parameters to regularize the problem, replacing the ill-posed problem by one that is well-posed, and thus has an acceptable solution. Finding and selecting good regularization parameters can be very expensive and subject to bias. Researchers often have invaluable information that is crucial in finding a good approximate solution, but without validation, there is risk of seeing what is expected and not the true solution or image. An effective automated tool that generates a plausible range of values is needed to create both a cost effective methodology and control for bias when determining optimal solutions.

2 Approach

The goal of this project is to develop a software package with graphical user interface (GUI) for parameter selection in regularizing ill-posed problems applied to debluring and denoising problems with Gaussian noise. The software will use generalized cross-validation (GCV) for initial parameter selection and statistical diagnostics to validate candidate solutions based on user provided parameters. A number of regularization methods will be used in hopes of determining optimal methods for a given image. To start, the software will include Truncated SVD (TSVD) regularization, Tikhonov regularization, and Total Variation (TV) regularization.

2.1 Regularization methods for ill-posed problem

Consider the ill-posed and ill-conditioned discrete problem $\mathbf{Ax} = \mathbf{b}$ where $\mathbf{A}$ is a known $m \times n$ or Point Spread Function (PSF) (blurring function in image applications) where $m \geq n$, $\mathbf{b}$ is a known $m \times 1$ vector (measured image) and $\mathbf{x}$ is unknown $n \times 1$ vector. In general to solve the inverse problem one would solve $\mathbf{Ax} = \mathbf{b}$ or the equivalent least square problem $\min \|\mathbf{Ax} - \mathbf{b}\|_2^2$ but because $\mathbf{A}$ is ill-conditioned solving $\mathbf{Ax} = \mathbf{b}$ directly generally does not give good results. To deal with the ill-posedness one needs to apply a method which imposes stability to the problem while retaining desired features of the solution. Regularization methods incorporate a priori assumption about the size and smoothness of the desired solution.

$$\min \|\mathbf{Ax} - \mathbf{b}\|_\mathbf{2}^\mathbf{2} + \gamma\mathbf{\Omega}(\mathbf{x})$$

Where the second term of the expression above is the regularization term where $\Omega(x)$ is smoothing function or penalty function and γ is the regularization parameter.

- Tikhonov's Regularization method:
  $\Omega(x) = \|\mathbf{Lx}\|_\mathbf{2}^\mathbf{2}$ and $\gamma = \lambda^2$
  Where $\mathbf{L}$ is the identity matrix, approximation of the first derivative operator, a diagonal weighting matrix [4], or any other type of operator based on the problem and the desired features.

- Total Variation:
  $\Omega(x) = TV(\mathbf{x}_\lambda)$ and $\gamma = \lambda$
  where $TV(\mathbf{x}_\lambda) = \int_\Omega \sqrt{|\Delta \mathbf{x}_\lambda|^2} d\Omega$.
  Where the penalty function is of norm l_1 instead of the norm l_2 like in Tikhonov regularization method which assumes that the desired solution x is smooth.

- Truncated SVD:
  In truncated SVD instead of smoothing out the high frequency like in Tikhonov we regularize the problem by truncating A and therefor ignoring the small singular values, the regularization problem becomes: $\min \|\mathbf{A_k x} - \mathbf{b}\|_2^2$ where $A_k = \sum_{i=1}^k u_i \sigma_i v_i^T$ where the regularization parameter is k or the level of truncation.

2.2 Statistical Based Diagnostics

The use of residual periodograms for choosing regularization parameters was demonstrated by Bert Rust and Dianne O'Leary [7] as an effective diagnostic tool to determine plausible regularized solutions. In this project we will use the following three diagnostic to generate a range of plausible regularization parameters.
Assuming that the errors in the data are independently identically normally distributed with mean zero and variance one, the discretized linear regression model is $\mathbf{b} = \mathbf{A}\mathbf{x}^* + \epsilon$, where $\epsilon \sim N(\mathbf{0}, \mathbf{I_m})$. Now if we have an estimate $\tilde{\mathbf{x}}$ of $\mathbf{x}^*$ then the residual vector $\tilde{\mathbf{r}} = \mathbf{b} - \mathbf{A}\tilde{\mathbf{x}}$ for a plausible $\tilde{\mathbf{x}}$ should be a sample from the distribution from which ϵ is drawn since our linear regression model can be written $\epsilon = \mathbf{b} - \mathbf{A}\mathbf{x}^*$. This characteristic of the residual inspired three diagnostics [7]:

- Diagnostic 1. The residual norm should be within two standard deviations of the expected value of $\|\epsilon\|_\mathbf{2}^\mathbf{2}$ or $\|\tilde{\mathbf{r}}\|_2^2 \in [m - 2\sqrt{2m}, m + 2\sqrt{2m}]$ where the expected value of $\|\epsilon\|_2^2$ is m and the variance is $2m$.

- Diagnostic 2. The graph of the elements of the vector of the residual $\tilde{\mathbf{r}}$ should look like samples from the distribution $\epsilon \sim N(0, 1)$. Quantitatively this test is based on the goodness of fit of the normal curve to the histogram of the elements $\tilde{r}_i$.

- Diagnostic 3. If we consider the elements of ϵ and $\tilde{\mathbf{r}}$ as time series with index $j = 1, ..., m$ where $\epsilon_j \sim N(0, 1)$ which form a white noise series then $\mathbf{r}$ should also be a white noise series. To measure this quantitatively one needs to find the cumulative periodogram of the residual time-series where when plotted is the partial sums of the periodogram versus the spectral density of the residual on the interval $[0, \frac{1}{2T}]$ and test to see if the plot falls within a 95% confidence band of the cumulative periodogram for the time series of white noise [7]. Where the periodogram is found by taking the discrete Fourier transform of the time series and taking the sum of the square the real and imaginary parts of the transform.

2.3 Total Variation (TV) Regularization

Several iterative methods for solving the minimization of the function above have been suggested including time marching schemes, steepest descent, Newton's method, lagged diffusivity fixed point iterative method among others [9]. The method and algorithm for solving the above equation will be determined as part of this project. The implementation of the algorithm will be done in the format of existing RestoreTool function and a effort for optimized coding and efficient memory allocation will be made.

2.4 Generalized cross-validation (GCV)

The method of generalized cross-validation (GCV) is to minimize the GCV function $G(\lambda) = \sum_{k=1}^{m}[b_k - (\mathbf{A}\tilde{\mathbf{x}}_\lambda^{(k)})_k]^2$ where the $\tilde{x}^{(k)}$ is the estimate when the k^{th} measurement of $\mathbf{b}$ is omitted. This method will be used to initially choose λ for each of the regularization methods [7].

2.5 Software

- Tikhonov and TSVD for images- already exist (part of RestoreTool [5])
- Generalized cross validation (GCV) for Tikhonov and TSVD regularization - already exist (part of RestoreTool [5])
- TV regularization to be added to RestoreTool [5]- to be built by Brianna Cash in Matlab using RestoreTool
- GCV for TV regularization-to be adapted from RestoreTool[5] GCVforSVD by Brianna Cash
- Statistical diagnostics - already exist (Dianne O'Leary)
- GUI interface- to be built by Brianna Cash using Matlab GUI tool box
- The software package will be written with the intent of running jobs in parallel using the MATLAB Parallel Computing tool box.

2.6 Hardware

For the initial stages the software will be designed to run on a modern desktop PC or laptop with no special hardware requirements. For the parallelization stage I will use available computers on either the computer science, or math-network that have multi-core machines available.

3 Databases

For this project I will use artificial generated images and PSF functions for development as well as some initial testing of the software. These data sets will be created to any size that is both menagebale and required. For testing and validation I will use the five test data sets in RestoreTool which include a 256X256 image and a least one test PSF.

4 Validation and Testing

At each step of the project I will need to verify that the results generated are working as the algorithms intended. I will use several test data where the point spread function and the original clear image are known in order compare the results to the original clear image for each algorithm implemented. When available I will compare my generated results for a specific test problem to published results.

In particular for this project I will need to validate the implementation of the total variation (TV) regularization and the generation of valid range parameters. For the implementation of TV regularization I will do this in several ways, first I will use simple binary test image problems with a known Point Spread Function (PSF) where the method given a good regularization parameter should return close to the original image (as measured visual and by the root mean square average magnitude of the errors in the elements of the estimate). Second, once

the method is chosen numerical analysis will be done to find the expected convergence rate and compare to my implementation. Third, I will use the test problems in RestoreTool with the suggested PSF as additional validation that the TV regularization produces expected results on larger problems for a variety of parameters.
For the generation of the parameters using generalized cross validation (GCV) and the three diagnostics the validity of the implementation will be based on the visual results and comparing the root mean square average magnitude of the errors in the elements. I will also compare the results of GCV and the range the three diagnostics produce.

I will assume that the code used from RestoreTool has been tested and validated. In addition to validating results of the code, the code will be measured for the amount of time and cost (memory) and compared to the expected time and cost for the algorithm.

5 Schedule

Fall Semester:

- Sept-Oct 15 - Learn to use RestoreTool and image processing tool box in Matlab
- Sept-Oct 15 - Learn to use Matlab GUI editor
- Oct 15-Nov 15 - Add diagnostics to Restore Tool and validate results
- Oct 15-Nov 30 - Read literature about Total variation, determine iterative method to be used in software
- Nov 30 - Dec 15 - Outline algorithm for total variation and start building basic total variation regularization in Matlab.
- Dec 1-Dec 15 - Prepare midyear report and presentation
- Dec 15-Jan 1 - Develop GUI for RestoreTool or Regularization Toolbox including diagnostics
- Nov 15-Jan 15 - Read literature about parallel computing

Spring Semester:

- Jan 15-Feb15 - Implement Total Variation in RestoreTool Framework
- Feb 15-Feb 28 - Validate Total Variation tool
- Mar 1 - Mar 15 - Add Total Variation tool to GUI
- Mar 15-April 30 - Optimize Total Variation using tool using parallel toolbox in Matlab (if available), if not look at other ways to implement the software package using other optimization the package as whole available in matlab.
- May - Prepare final report and presentation

6 Milestones

- Oct 15 - A basic GUI to use existing tools in RestoreTool [5] or Regularization Tool [3]
- Nov 30 - Add validated periodogram diagnostics to RestoreTool
- Dec 1 - GUI that determines a range of plausible regions using the periodogram diagnostics
- Dec 15 - Outline of Total Variation regularization algorithm and basic implementation in Matlab.
- Dec 15 - Deliver mid-year report and presentation
- Feb 1 - Total Variation (TV) regularization in RestoreTool Framework
- Feb 15 - Generalized Cross Validation (GCV) for Total Variation tool
- Feb 30 - Add validated TV too with GCV to the GUI
- April 1 - Optimize software package finished (using parallel toolbox in Matlab if available)
- May 1 - Poster for SIAM Conference on Image Science to be presented May 20-22
- May 15- Deliver final presentation and report

7 References

1. R. Acar and C.R. Vogel. Analysis of Bounded Variation Penalty Methods for Ill-Posed Problems. *Inverse Problems.* 10: 1217-1229, 1994.
2. Tony F. Chan and Jianhong Shen. Image Processing and Analysis. *SIAM*, Philadelphia, PA, 2005.
3. Per Christian Hansen. Regularization Toolbox, http://www2.imm.dtu.dk/ pch/Regutools/regutools.htm
4. Per Christian Hansen. Rank-Deficient and Discrete Ill-Posed Problems: *Numerical Aspects of Linear Inversion. SIAM*, Philadelphia, PA, 1998.
5. James G. Nagy, RestoreTool, http://www.mathcs.emory.edu/ nagy/RestoreTools/
6. James G. Nagy, K. Palmer and L. Perrone. Iterative Methods for Image Deblurring: A Matlab Object Oriented Approach. *Numerical Algorithms.* 36: 73-93, 2004.
7. Bert W. Rust and Dianne P. O'Leary. Residual programs for choosing regularization parameters for ill-posed problems. *Inverse Problems*, 24:034005 (30 pages), 2008. Invited Paper
8. Bert W. Rust. Parameter selection for constrained solutions to ill-posed problems. *Computing Science and Statistics*, 32:333-347, 2000.
9. C. Vogel and M. Oman. Iterative methods for total variation denoising, *SIAM Journal on Scientific Computing*, Vol.17, pp.227 238, Jan.1996.